Using Intelligent Load Adjustment to Find Feasible Power Flows in Emergency Situations

by

Edward Nguyen

B.S., Massachusetts Institute of Technology (2019)

Submitted to the Department of Electrical Engineering and Computer Science

in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2020

Author . . . .
Department of Electrical Engineering and Computer Science
September 2020

Certified by . . . .
Marija Ilic, Senior Research Scientist
Thesis Supervisor

Accepted by . . . .
Katrina LaCurts
Chair, Master of Engineering Thesis Committee

Using Intelligent Load Adjustment to Find
Feasible Power Flows in Emergency Situations

by

Edward Nguyen

Submitted to the Department of Electrical Engineering and Computer Science
on July 31, 2020 in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Economic forces guide power generation and distribution in power grids. In natural disasters and other emergency scenarios transmission lines can become overloaded and fail or power companies may preemptively blackout neighborhoods to prevent cascading failures. Both scenarios cause end users to lose power unnecessarily because the power market cannot create a feasible solution fast enough to avoid these negative outcomes.

This thesis presents an adapted max-flow algorithm as part of a protocol that schedules power flows during an emergency. The power flow assignments fall within network constraints such as thermal limits of transmission lines. The algorithm assumes adjustability of load demand and allocates power to loads following the max-min fairness rule.

We implement and evaluate this protocol on the IEEE 118 Bus dataset subjected to a number of emergency scenarios. We benchmark the speed of the algorithm against previous max-flow approaches to power grid resiliency and we measure the efficacy of the algorithm by evaluating its ability to supply a critical load percentage to each load bus.

Thesis Supervisor: Marija Ilic
Title: Senior Research Scientist

Acknowledgments

Marija, for her endless guidance, patience, and advice.

Rupa, for helping me understand the fundamentals of the problem space and field that I was not familiar with.

Dr. Reidl, Dr. Stewart, Max, Ian, Eran, Mark and the long list of mentor figures and teachers who have brought me to where I am today.

Matt, David, Ben, Anna, Laurel, and the rest of my living group in Simmons who have kept me grounded throughout our years at MIT.

My siblings, Glory, Kai, Sophia, Antonin, Erik, and Alexander for inspiring me to persevere at my lowest points.

And finally my parents, Trung Nguyen and Ngoc Chau T. Thai, for their decades of hard work to provide me with the opportunities I've had and for their unwavering support.

Contents

List of Figures

Chapter 1

Introduction

Power grids and infrastructure are ubiquitous across the United States. The average person travelling through the country will see endless miles of transmission lines and numerous substations and plants dedicated to supplying power to the roughly 330 million people within the country.

The Department of Energy has recognized a number of vulnerabilities in the power grid including natural disasters, terrorism, accidents, and sudden supply/demand imbalances [8][9]. Many of these vulnerabilities are systemic such as cyberterrorists hacking into the grid and disabling a number of power plants; but others are not such as a tornado ripping through a swath of power lines. This thesis deals with improving the robustness of the grid in the latter scenario.

In a non-emergency scenario power flows are dictated by economic forces. Customers and suppliers bid on power generation and the market guides efficient power generation and consumption. This process is called Economic Dispatch and is solved through algorithms such as Direct Current Optimal Power Flow (DC-OPF). Convergence of these algorithms, however, is too slow to be useful in emergency scenarios [2].

Consider now, an emergency situation: a piece of a power grid is disabled or destroyed. Existing control mechanisms will try to dynamically adjust power distribution in an attempt to satisfy the remaining load. This can cause a higher steady state power flow in some power lines than the cables are rated to carry. This load causes the cables to heat up before eventually failing, putting additional stress on the rest of the infrastructure and creating a cascading failure.

In these circumstances, a topological analysis of the remaining infrastructure can reveal a redistribution of power flows along transmission lines to satisfy current load without overloading any one line and identify when such a distribution does not exist [1]. When a distribution does not exist we do not have automated solutions in place.. Instead, timely power plant managers may reduce the power load supplied to various areas in so-called "brown-outs" or "black-outs" in order to protect critical infrastructure with a priority on preventing a cascading failure. This reduction in load can have negative societal impacts such as turning off life-saving medical equipment within peoples homes [5].

Only a fraction of power consumption during an emergency is really necessary such as hospitals, home medical equipment, communication systems, emergency services, etc. That portion of power forms what we will refer to as a "critical load" and that should be satisfied if possible. The problem is that we may not know what this critical percentage is at any given time. Furthermore, it may be harmful to outright measure it for two reasons. One, a formal designation of infrastructure as within the "critical load" opens security risks in allowing bad actors to optimize damage done in a terrorist attack on the power grid. Second, it would incentivize individuals to exert pressure to be included in that "critical load" during an emergency.

This thesis presents an algorithm to attempt to solve this problem. In the event of an emergency where the current infrastructure is incapable of supplying the current load demanded the algorithm will suggest a new set of adjusted generation, distribution, and load that will prevent cascading failures and attempt to satisfy critical loads as well as possible. No direct information about the fraction of normal, or "steady state", load that is "critical" will be used in order to sidestep the security risk and prevent creating perverse incentives.

The core innovation is the application of resource allocation ideas to previous research. Specifically we apply the max-min fairness rule common in networking to power distribution in order to even out the power supplied to different end users [10]. This operates under the hypothesis that critical loads are a fraction of total load and equitable distribution of power resources will lead to satisfying the most of these critical loads.

1.1 Summary

This thesis presents a theoretical algorithm for equitably distributing power generation to load buses during an emergency to maximize the critical proportion of steady state loads that are satisfied while abiding by network constraints.

The rest of this thesis is organized as follows. Chapter 2 covers the previous conducted work related to this problem that inspired this approach. Chapter 3 formalizes the Max-Min Power Flow Algorithm and analyzes its theoretical runtime and properties. Chapter 4 discusses the metrics used to evaluate the protocol's effectiveness along with a general description of simulation design and implementation. Chapter 5 provides the results of the simulations described in Chapter 4 along with analysis of the results. Chapter 6 discusses the conclusions of this thesis and the protocol's effectiveness. Chapter 7 suggests a number of extensions to build on top of this thesis's work. Finally a bibliography is included with sources along with an appendix containing complete tabulated simulation results.

1.2 Contributions

The contributions of this thesis are:

- **Critical Load Percentage:** A construct with which to evaluate the effectiveness of a set of adjusted power flows in an emergency scenario.
- **Max-Min Power Flow**: An algorithm designed to assign flows in an emergency to provide power effectively without overburdening the transmission infrastructure.
- **Protocol:** An example of how the Max-Min Power Flow algorithm can be applied to manage emergency situations.
- **Implementation**: A sample implementation of the algorithm with simulations to measure is efficacy and efficiency.

Chapter 2

Background and Related Work

2.1 Critical Load

In discussing reliability of power systems it is common to discuss the concept of a "critical load". Generally, this refers to a bus within a power grid that is deemed "critical" and whose power consumption should be prioritized over "non-critical" buses. Within this thesis, however, every load bus will have some amount of "critical load". The proportion of total load that is critical will be referred to as the **critical load percentage.** Figures 1 and 2 below provide a simple example illustrating this concept:

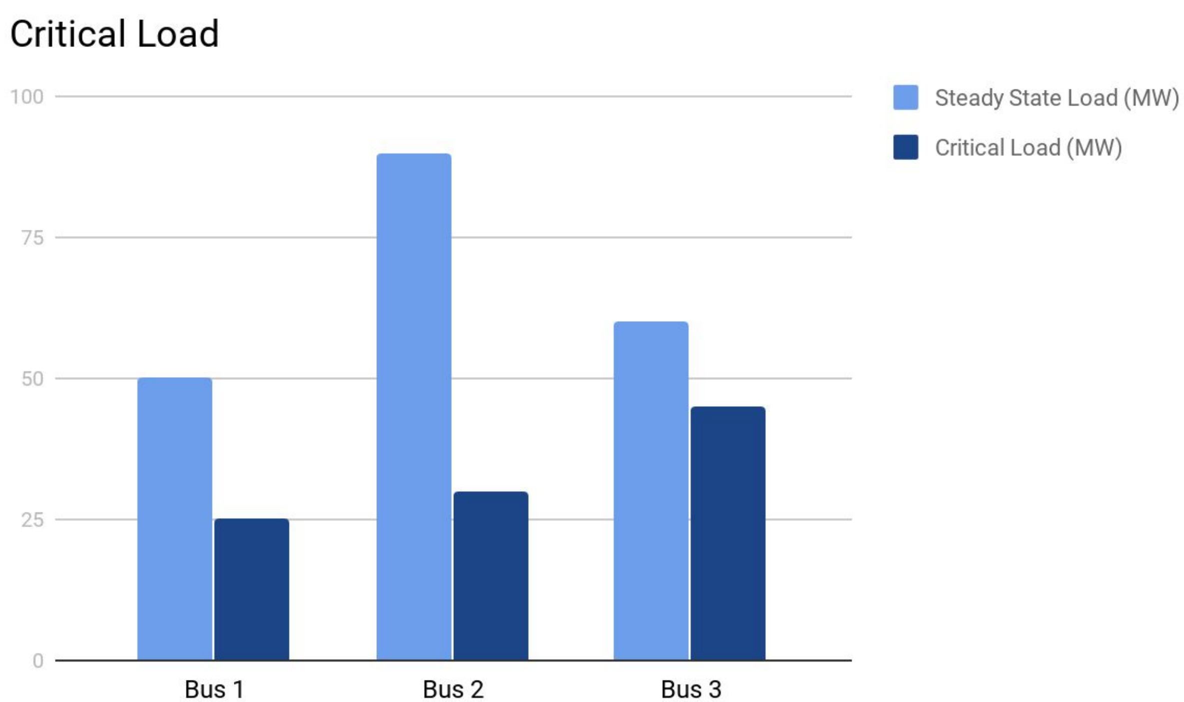

Figure 1: Graph showing sample critical loads against steady-state loads.

Bus	Steady-State Load (MW)	Critical Load (MW)	Critical Load Percentage
1	50	25	50%
2	90	33	33%
3	60	45	75%

Figure 2: Table with specific values shown in Fig. 1

2.2 Graph Theory in Power Grid Design

Power generation and transmission within a power grid can be represented as a directed graph with power flowing from vertices representing generators to those representing consumers [11]. Each bus within the network is represented by a unique vertex within the graph with edges representing the existence of transmission lines connecting those vertices and an assigned magnitude corresponding to the transmission capacity of that line. Power flowing from one bus to the next can be represented by a flow along a directed edge that is less than or equal to the capacity of that edge.

Generation can be simulated by attaching each generator bus to a universal source by an edge whose capacity is equivalent to the generation capacity of that bus. Similarly, load can be simulated by attaching each load bus to a universal sink by a directed edge whose capacity is equivalent to the current drawn load. Figure 3 shows an example network and Figure 4 shows how that network would be represented as a graph.

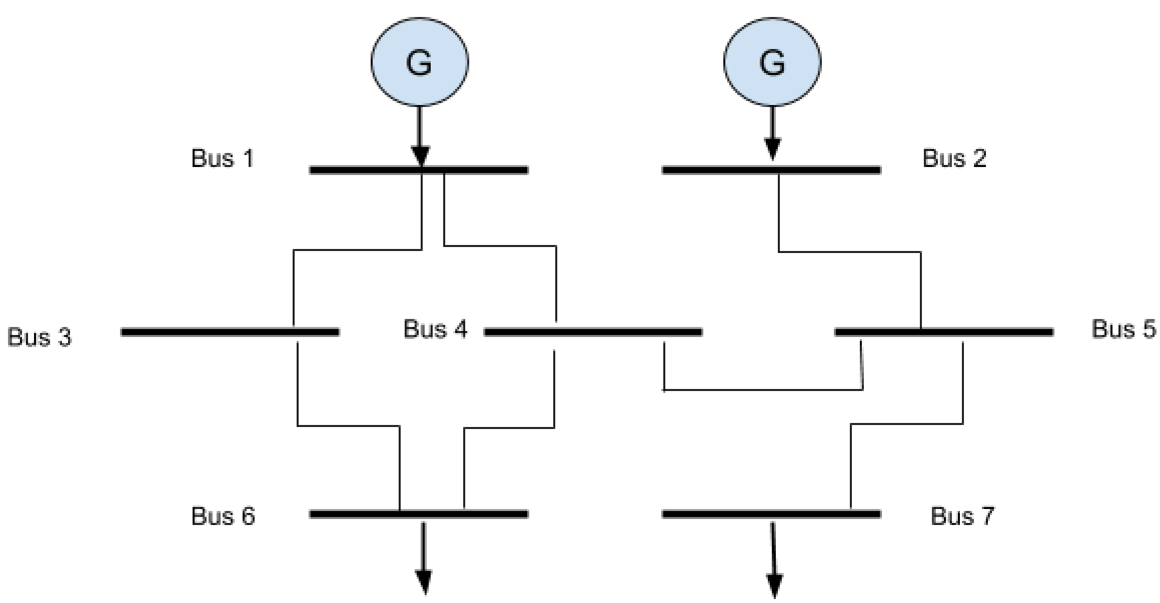

Figure 3: Sample system diagram of a 7 Bus, 2 Generator, 2 Load System

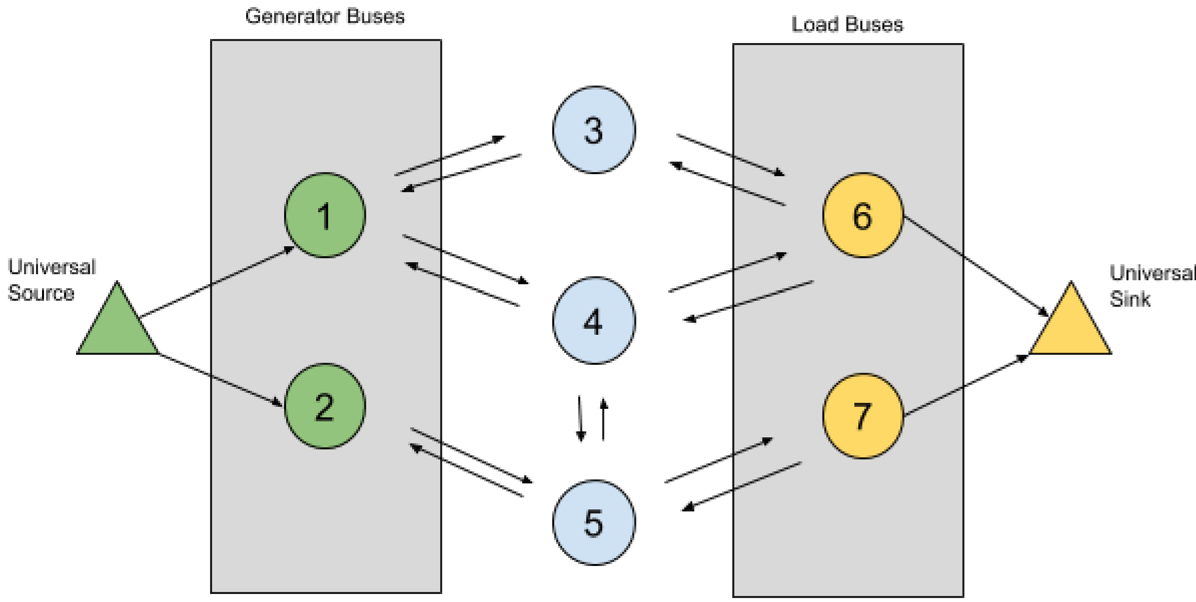

Figure 4: Equivalent graph representation of Figure 3. Yellow nodes represent loads or the sink, green nodes represent generators or the source and blue nodes represent distribution nodes.

2.3 Distributed Max Flow Algorithms in Power Grid Design

In 2005, researchers showed that Max Flow algorithms could be useful in preventing cascading failures or at least predict their occurrence during an emergency. Using a distributed version of Goldberg-Tarjan's push-relabel max-flow algorithm, the researchers were successful in creating an algorithm that could reassign power flows to avoid cascading failures or detect that no such reassignment existed. The distributed version of this algorithm worked in seconds providing ample time for the necessary adjustments to take place [1].

In the event of an inevitable failure, however, the paper suggests load shedding as a last resort, but does not touch on how that shedding should be done. This paper builds directly on this work to suggest an effective method for adjusting loads.

2.4 Max-Min Fairness and Progressive Filling

Like power grids, computer networks can also be modelled by directed graphs. In this case, vertices represent endpoints for messages such as servers and computers while edges represent direct connections between those entities. These networks only have a certain amount of bandwidth that may not be enough to satisfy the data transfer demands of all its end users so establishing fairness is an issue. Similarly, during an emergency, it is in the communal interest that power be distributed fairly to ensure that as much critical consumption as possible remains satisfied.

The max-min fairness (MMF) rule is one idea within networking to reach a fair allocation of bandwidth. The rule can be summed up as "all entities are assigned an amount of resources such that the increase of any resource allocation necessitates the reduction of at least one other resource allocation of lesser or equal value". In more colloquial terms, no one can be assigned more resources unless those with less are unable to receive more, typically due to capacity constraints of the network topology [10].

Progressive filling is an algorithm designed to achieve MMF within networks. The core concept is to slowly increase the bandwidth assigned to all individuals. When an individual's bandwidth cannot be increased due to network constraints, they are considered satisfied and only the remaining individuals' bandwidths are incremented [7].

Chapter 3

Max-Min Power Flow

This section enumerates an algorithm designed to distribute power flow equitably across a set of loads given a network topology, generation capacities, and current load demands. The section includes a simple protocol for implementing this algorithm for managing power flows during an emergency.

Fundamentally, the goal of the algorithm is to provide, for a set of load buses, a set of power allocations that are MMF. Formally, let $\mathbf{x}$ be a vector of power allocations to a set of load buses. Let $\boldsymbol{X}$ the set of such vectors that are feasible given network constraints and the steady state load demands such that $\boldsymbol{x} \in \boldsymbol{X}$. The max-min fair allocation will be equivalent to
the lexicographical maximum of set $\boldsymbol{X}$ [10].

We find this maximum by solving a series of convex optimizations problems. Specifically we split the set of buses into two sets: $\boldsymbol{B}$ and $\boldsymbol{B'}$. Let $\boldsymbol{B}$ be the set of indices in $\boldsymbol{x}$ for which the steady state load demanded by bus i is greater than or equal to that of any element of $\boldsymbol{B'}$. Let $\boldsymbol{t}$ represent the results of solving the maximization problem for previous splits of $\boldsymbol{B}$ and $\boldsymbol{B'}$. We then solve

$$
\begin{array}{lll}
\textit{maximize} & \tau & \\
\textit{subject to} & \boldsymbol{x_k} \geq \tau & \boldsymbol{k} \in \boldsymbol{B'} \\
 & \boldsymbol{x_k} \geq \boldsymbol{t}_{\mathrm{k}} & \boldsymbol{k} \in \boldsymbol{B} \\
 & \boldsymbol{x} \in \boldsymbol{X} &
\end{array}
$$

After solving the equation, the values of k for which $\boldsymbol{x_k} = \tau$ are added to $\boldsymbol{B}$ and $\boldsymbol{t}_{\mathrm{k}} = \tau$ and the problem is solved again with a new set $\boldsymbol{B}$. The rest of this chapter is dedicated to enumerating the details of how this is achieved using a graph representation of a power grid and a Max Flow algorithm to solve the optimization problem.

3.1 Definitions

- Let V represent the buses within a given power grid.
- Let V_g represent a subset of V consisting of the buses within V that produce power.
- Let V_l represent an ordered subset of V consisting of the buses within V that consume power sorted in increasing order of load demand. $V_{l,n}$ represents the bus with the nth lowest load within the subset V_l for $0 \leq n < |V_l|$
- Let $c(u, v)$ and $f(u,v)$ be mappings $V \times V \rightarrow R^+$ representing the capacity and flow of directed edge (u, v) respectively.
- Let E represent the connected transmission lines between the buses of V within a given power grid.
- Let s and t be the universal source and sink, respectively
- Let $g(u)$ be the maximum generation capacity of u subject to $u \in V_g$
- Let $l(u)$ be the current power load of u subject to $u \in V_l$

3.2 Algorithm Description

The crux of the algorithm involves performing a max flow analysis on the power grid. We attach a directed edge from the universal source to each of the generators with a capacity equal to the maximum generation capacity of that generator. We also attach a directed edge from each consumer to the universal sink.

1. Let G be a graph representing the power grid **(V, E)**.
2. Let G^+ be an augmented graph (V^+, E^+) where $V^+ = V \cup \{s, t\}$ and $E^+ = E \cup \{(s, u): u \in V_g\} \cup \{(u, t): u \in V_l\}$. The capacity $c(s, u) = g(u)$ for $u \in V_g$ and $c(u, t) = l(V_{l,0})$ for $u \in V_l$
3. Perform any max flow implementation of the Ford-Fulkerson method, or any method that also produces a residual network on G^+
4. If the calculated max flow is equal to the capacity of the edges directly connected to the universal sink (hereafter referred to as the load edge subset) set $c(u, t) = l(V_{l,n+1}) - l(V_{l,n})$ for $u \in V_l$ where n was the previous index of V_l used to set the load edge subset capacities, set the capacities of the remaining edges within G^+ to the residual capacities as defined inside the residual network of G^+. Remove edge $(V_{l,n-1}, t)$ from E^+ and repeat step 3.
5. Take the max-flow calculated during step 4 and divide it by the size of the remaining load edge subset. Set the capacity for each edge to this value and the capacities of the rest of the grid to the residual network of G^+. Rerun the Max-Flow implementation on G^+.

6. For any flow along the load edge subset that is less than its capacity, remove the edge. Repeat Step 5 with the max-flow calculated in step 4 replaced with the max-flow currently calculated minus the flows of the removed edges. When the load-edge subset is empty, stop.

The flows within the generation edge subset represents the generation each power plant should be set to; the flows within the load edge subset represent the MMF vector $\boldsymbol{x}$ representing power allocations to load buses; The flows within the rest of the edges represents a feasible assignment of power transmission that will satisfy the adjusted loads.

3.3 Proof of Correctness

3.3.1 Power allocation is feasible

A feasible allocation is any allocation where the assigned power flowing down any transmission line and the assigned generation of any bus is less than or equal to its respective capacity. Within the context of the graph this is equivalent to showing that the flow along any edge is less than or equal to that edge's capacity. Flow amounts are assigned by an execution of a max-flow algorithm that respects edge capacity, so all flow assignments from this algorithm are feasible.

3.3.2 Max-Min Fairness Rule upheld

Lemma 1: Edges removed from the load-edge subset are removed in monotonically non-decreasing order of final assigned flow.

Each time an edge is removed it is always the edge with either the lowest capacity (in step 4) or the lowest flow (in step 6) of the remaining edges within the load-edge subset. Each iteration of the max-flow algorithm is applied to the residual graph after allocating a set amount of power flow to each remaining edge. These allocations are set and since additional power flows are greater than or equal to zero, each load edge, when removed, must have an allocated power flow equal to or greater than that of the load edge that was removed previously.

We will now show that the MMF rule is upheld within the algorithm by proof by induction. By Lemma 1 we only need to show that the MMF rule applies to the removed edges from the load edge subset upon the removal of any given edge since future removed edges will have greater power flow allocations.

Base Case: First edge is removed
Trivially true since every other edge will have a greater allocation.

Inductive Case: An edge is removed with a non-empty removed subset.
By Lemma 1 we know that the flow assignment to the edge must be greater than or larger than the flow assignments to the edges that preceded it. We also know that the edges removed prior to this edge form a subset that abides by the MMF rule.

If the edge was removed during step 4 then the load demand is completely satisfied and it is impossible to increase the power allocation to that edge.

If the edge is removed during step 6 then we must show that the flow allocated to the edge cannot be increased by decreasing the flows of larger power allocations. Note that the power allocations for each load is capped at the average max flow. Every load that hits that artificial cap represents a bus that more power can be assigned to. Load edges that do not hit that flow capacity are thus maxed out and cannot be assigned more power even if the other flows assignments are reduced.

3.4 Runtime Analysis

The asymptotic runtime of the algorithm is dependent on the choice for max flow subroutine. Within this thesis I will use Dinic's Algorithm which has an asymptotic runtime of $\mathbf{O(|V||E|\log|V|)}$ [15]. Note that for each subsequent execution of the subroutine at least one edge is removed from the load-edge subset. Taking this into account along with the sorting of $\boldsymbol{V_l}$ we have an asymptotic runtime for the Max-Min Power Flow of $\mathbf{O(|V|^2|E|\log|V| + |V|\log|V|)}$. Note that power grids are planar and thus **G** is a sparse graph, implying that **|E|** is proportional to **|V|** giving us a final asymptotic runtime of $\mathbf{O(|V|^3\log|V|)}$ [4].

3.5 Emergency Protocol

We assume that before the emergency event, the power grid is in a steady state where power generation and transmission capacity are sufficient to satisfy current load demand. While in this "steady" state, a real time model of the power grid is maintained with up-to-date loads, generation, and power flows.

Protocol:

1. While in “steady” state maintain a model of the network and monitor transmission line and generator status.
2. When the power output of a generator suddenly declines without scheduling or a line is broken, enter an “emergency” state.
3. Update the topology of the saved network model to reflect sudden changes.
4. Run a max-flow analysis on the updated network.
5. If the infrastructure is sufficient to satisfy demand, allow it to reach a new equilibrium. Exit “emergency” state and return to “steady” state.
6. If not, run the Max-Min Power Flow algorithm on the network. Adjust generation, transmission flows, and loads appropriately.

After a period of time at this allocation, relax restrictions and allow economic forces to push the grid to a new equilibrium state. Exit “emergency” state and return to “steady” state.

Chapter 4

Evaluation Strategy

4.1 Dataset

We use the public IEEE 118 Bus system as the dataset to evaluate our algorithm. Since we are modelling emergency scenarios and natural disasters it is important that our dataset represents a large system in order to test the robustness of the algorithm against a variety of failure conditions. The IEEE 118 Bus system is one of the largest, public, widely-available dataset and that is why it is chosen. Furthermore, this dataset has been used in the past to test other emergency protocols [1]. This allows us to benchmark our runtime against previous results using the same input. Figure 6 shows the system diagram for the 118-Bus system [12].

The Max-Min Power Flow algorithm requires four inputs to function: the topology of the grid, the steady state generator outputs, the steady state load demands, and the capacity of the edges. The first is directly enumerated by the dataset; The middle two can be calculated by using a OPF solver on the dataset; The latter can be set to either the emergency thermal limit (enumerated by the dataset) or the electrical limit (calculated from the reactance of the line).

4.2 Metrics

We are concerned with two questions: Can a max-min fair distribution be found fast enough to prevent a cascading failure? How well does that distribution satisfy critical loads? To answer these we use three different metrics.

4.2.1 Unit Critical Load Satisfied

In order for the critical load of a bus to be satisfied it must be supplied at least as much power as the critical percentage of its steady state load. After all, if you are on a dialysis machine that requires 30 Watts of power and we only supply 20 Watts the machine is still going to fail. We define Unit Critical Load to be the percentage of load buses whose critical percentage is satisfied in a power flow assignment.

4.2.2 Percentage Critical Load Satisfied

While the Unit Critical Load is important, it does not provide a holistic view of the effectiveness of a power flow assignment. It treats a power flow assignment providing 0% of critical loads to all buses the same as one providing 99% of critical loads. Percentage Critical Load is defined as the percentage of the sum of all critical loads that is satisfied by a power flow assignment. Figure 5 provides an example of how these values are calculated.

Bus	Steady State (MW)	Critical Load (MW)	Max-Min Load Assignment (MW)
1	80	30	25
2	60	15	25
3	20	15	20

Figure 5: Table showing an example load assignment to 3 buses.

Buses 2 and 3 have sufficient power allocated to satisfy their critical loads, but Bus 1 does not meaning this assignment has a Unit Critical Load metric of ⅔ or .6667. The total critical load is 60, but only 55 of that is satisfied giving a Percentage Critical Load metric of .9167.

4.2.3 Runtime

This metric is self-explanatory. Power lines have emergency thermal limit ratings on the order of hours so it is important that a satisfactory flow assignment is found in minutes or preferably seconds or faster [6]. Previous research was able to identify a satisfactory load assignment, or lack of one, in 5.72 seconds on average so our target runtime is on the order of seconds [1]. This allows for sufficient time for the system to adjust to the power assignment.

4.3 Critical Load Model

Critical loads are not, as of now, measured for consumer power loads. In the Future Work and Extensions section (Chapter 7) I go into further detail of how we could go about measuring this. For the purposes of evaluation we will assume that the critical load percentages are normally distributed around a mean value and observe the difference in the algorithm's performance over a range of means.

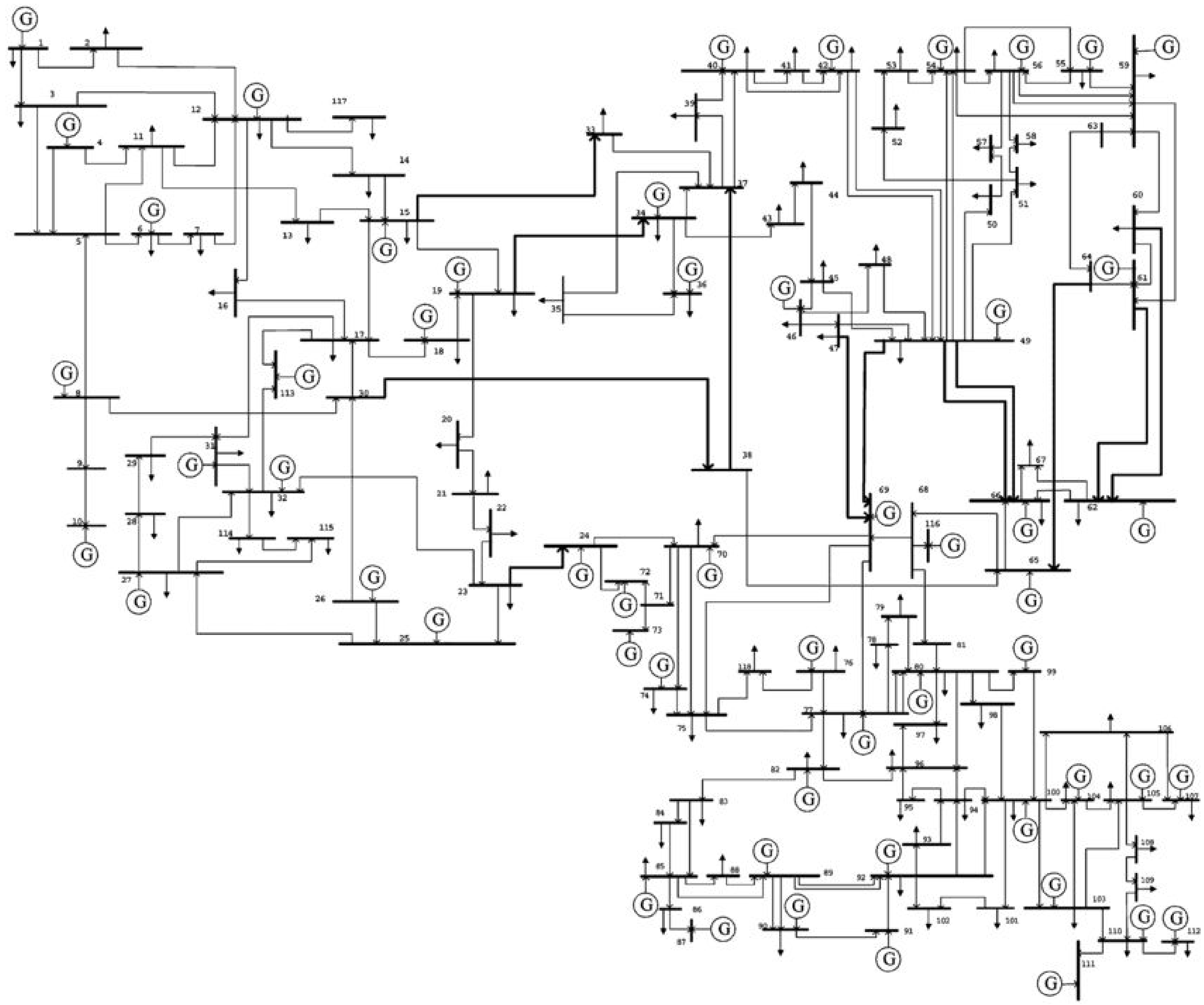

Figure 6: System diagram of the IEEE 118 Bus System

These randomly selected critical load percentages will be used to calculate the Unit Critical Load and Percentage Critical Load metrics for load assignments generated by the Max-Min Power Flow algorithm.

4.4 Simulation Design

When loads cannot be satisfied by the current state of the network there are two potential problems. One, there is insufficient power generation capability to satisfy the loads; Two, the capacity of the transmission network is not sufficient to satisfy the load demanded. This thesis approaches each problem separately to see how the Max-Min Power Flow algorithm performs in each situation individually.

Some elements are common between the two types of simulation. For example, the power generation during the steady state before an emergency is the same for both. I calculate this value using an OPF solver. Of the 54 generators in the 118 Bus system, only 38 generate power in the steady state. See section 10.1 within the appendix for details and exact values. Furthermore, both simulations require a random assignment of critical load percentages to the bus for truncated normal distributions with means ranging across the range of possible percentages (0 to 1). Since these assignments are random, the simulations must be run multiple times for each chosen mean in order to reduce variance. For both simulations, the chosen mean percentages are between 10% and 90% inclusive and at increments of 5%. The standard deviation of the distribution is constant at 5% and each simulation is run 100 times each.

4.4.1 Insufficient Power Generation

In the real world, what this could look like is a terrorist attack or a natural disaster takes out some number of generation plants such that the maximum generation potential of the remaining plants is lower than the current power consumption.

This scenario is modeled by selecting a random sample of the 38 active power generators in the steady state and reducing their generation capacity to 0. Line capacities are set to the thermal limits of the cables and disasters are simulated by removing 1, 4, or 8 generators simultaneously to represent varying levels of disaster intensity.

4.4.2 Insufficient Transmission Infrastructure

A portion of the transmission cables have been destroyed and the remaining cables cannot supply enough power for the remaining consumers still connected to the grid. This could be caused by a tornado ripping through the transmission lines or a car accident toppling a pole.

This scenario is modeled by making two alterations to the steady state network. First, the thermal capacity limits of the transmission lines within the 118 bus system are far too lenient. Any single line is easily capable of transmitting power equal to the magnitude of the total power requested by all buses in the system. The capacity of each line is set to its electrical limit instead, which can be estimated by the reciprocal of the reactance of the line [3] [13]. As it turns out, this alone reduces the maximal flow through the network to around 75% of the steady state requested load. The second alteration made is to select a random sample of the 186 lines connecting the 118 bus system and remove them from the network, disconnecting their buses. Disasters are simulated by removing 1, 4, and 8 power lines simultaneously to represent varying levels of disaster intensity.

4.5 Implementation Details

The protocol and simulations are coded in Python using the PyPower package, a python port of the widely used MatPower package used for analyzing power systems. This package is also the source of the OPF solvers used to generate the steady state generation and loads. The algorithm and simulations have been implemented almost exactly as described in Chapters 3 and 4, but a couple of optimizations have been used to improve the runtime of the simulations.

First, the results of the steady state generation output are identical at the start of each iteration. These values are cached and loaded at the start of each simulation iteration.

Second, while the formal description of this algorithm uses a Max-Flow implementation wholesale, this is unnecessary in implementation. When it is impossible to push more flow through the network, the result is checked, the loads are adjusted (as defined in Chapter 3) and the Max-Flow implementation resumes attempting to push more flow through the system. This has no effect on the theoretical asymptotic runtime, but potentially a modest effect on the experimental runtime.

Chapter 5

Results

Figures 7 and 8 show the Unit Critical Load and Percentage Critical Loads respectively of the Insufficient Power Generation Simulation described in Section 4.4.1. Figures 9, 10, 11, and 12 show these metrics for the Insufficient Transmission Simulation described in Section 4.4.2. Full numerical results for all simulations can be found in sections 9.2 to 9.4 in the appendix.

5.1 Insufficient Power Generation Simulation

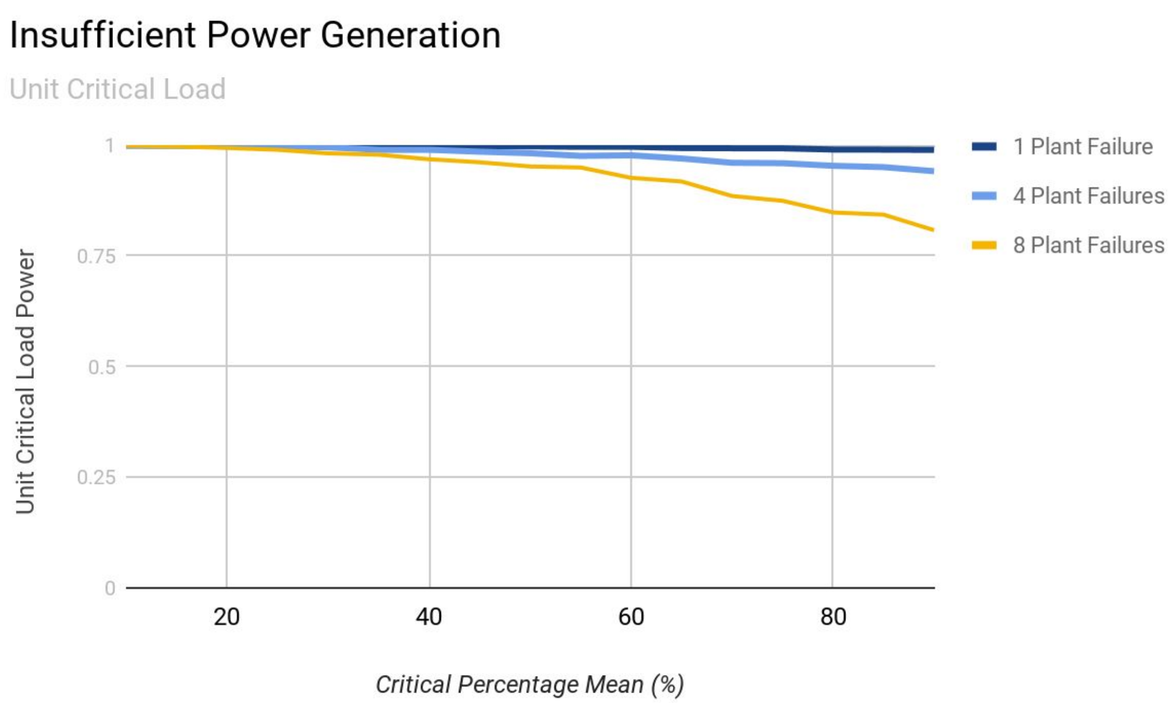

Figure 7: Unit Critical Load for the Insufficient Power Generation Simulation charted against the average critical percentages of load buses.

Figure 7 shows the Unit Critical Load Power metric of the Max-Min Power Flow algorithm for Critical Percentage Means ranging from 10% to 90% and for 1, 4, and 8 plant failures. Even in the worst case of 8 failures, which reduces the average power generation by 21%, the algorithm is able to redirect power flows such that over 75% of the buses have their critical loads satisfied. And for lower Critical Percentage Means like 40%, the algorithm satisfies the critical loads of over 90% of the buses.

This means that even for numerous simultaneous power generator failures, with excess transmission capacity the Max-Min Power Flow algorithm is able to satisfy the vast majority of critical loads until a longer, steady state solution can be reached.

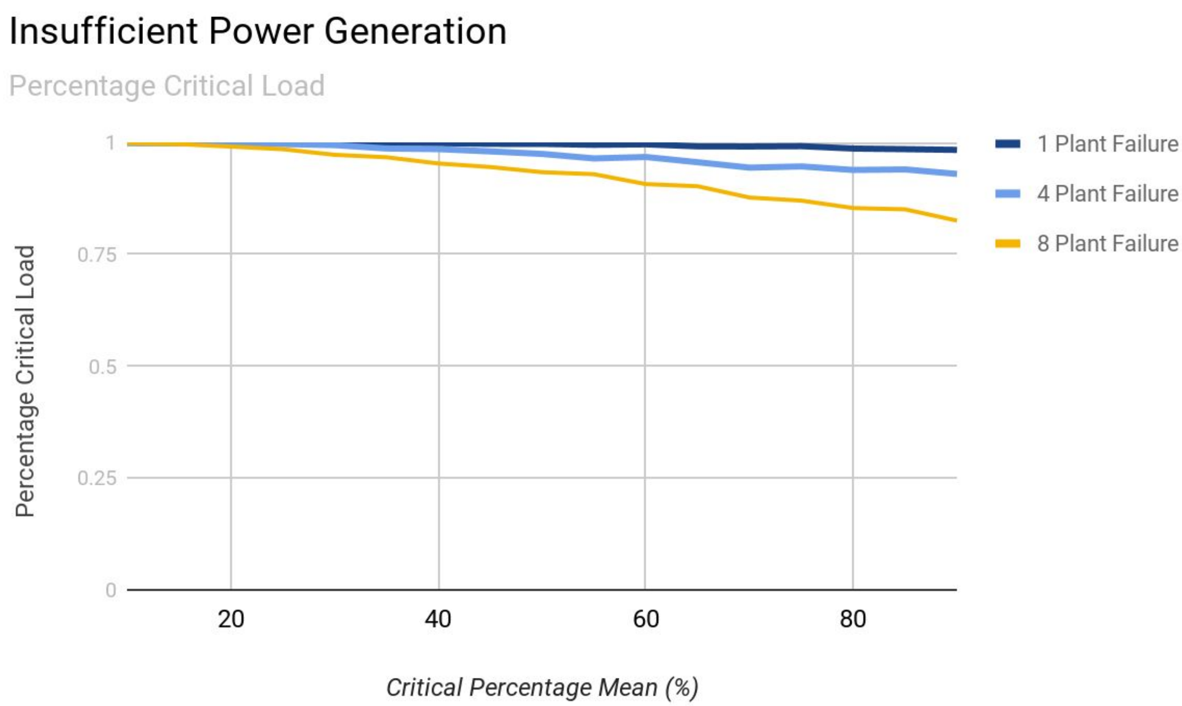

Figure 8: Percentage Critical Load for the Insufficient Power Generation Simulation charted against average critical percentage of load buses.

In Figure 8 we note a similar result in the Percentage Critical Load metric as in the Unit Critical Load. There is an earlier and more drastic separation in the 20-40% mean range where the Unit Critical Load is higher than the Percentage Critical Load. This is indicative of an unequal distribution of load demanded in the steady state. Lots of small loads are being prioritized and filled first inflating the Unit Critical Load, meaning a higher percentage of buses are getting enough power and those that aren't are demanding far more to begin with. Practically, this means that power usage cannot be manipulated to secure more power allocation in an emergency scenario, an ideal result.

5.2 Insufficient Transmission Infrastructure Simulation

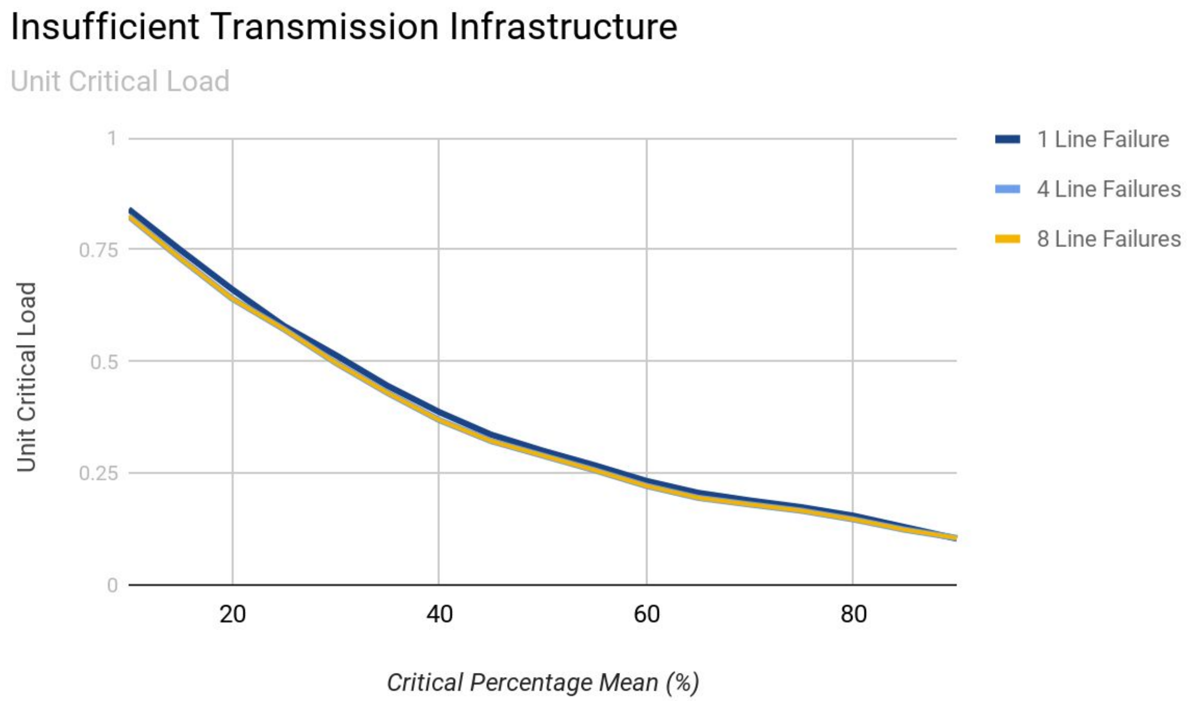

Figure 9: Unit Critical Load for the Insufficient Infrastructure Simulation charted against average critical percentage of load buses.

Figure 9 shows the Unit Critical Load for the Insufficient Transmission Infrastructure Simulation for Critical Percentage means ranging from 10-90%. Note that there is little separation between the 1, 4, and 8 line failures results. The change of the line capacities to the electrical limit as calculated from each line's reactance results in a network whose power generation and demand is much higher than the max flow of the network. In such a scenario there is little noticeable difference between the results produced by a different number of lines destroyed and the Max-Min Power Flow assignment reduces to a Max Flow assignment.

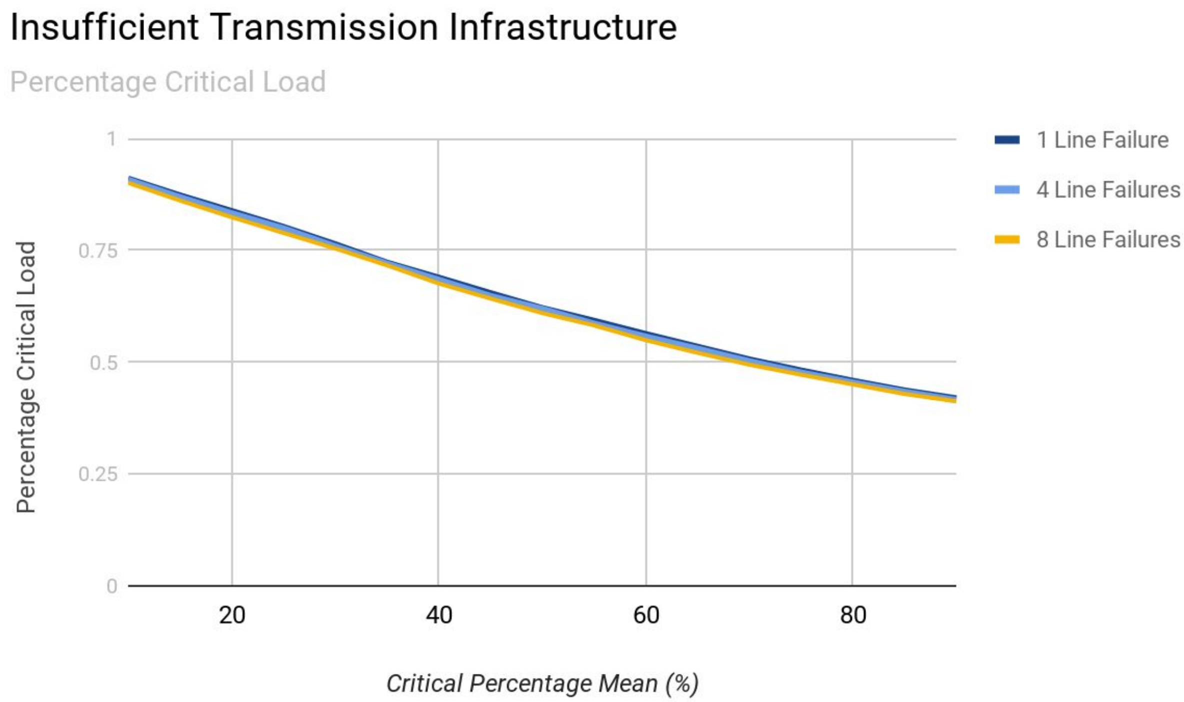

Figure 10: Percentage Critical Load for the Insufficient Infrastructure Simulation charted against average critical percentage of load buses.

Figure 10 shows us similar results as Figure 9. Flow capacity being unable to satisfy steady state power flows has reduced the assignment to a max flow algorithm leading to similar results regardless of disaster intensity. Also note that the line is almost linear. When all lines are at max capacity, the assignment is roughly the same regardless of which lines fail. This means that as one increases the critical percentage mean, the percentage critical load is proportionally decreased since nothing changes in the simulation results.

In order to test this we relax the capacity limits in the Insufficient Transmission Infrastructure. A new simulation is run where the calculated electrical limit is doubled, giving the grid the flow capacity to deliver all the power requested, but susceptible to bottlenecks.

The results of the Max-Min Power Flow algorithm are benchmarked against the Max-Flow algorithm results to check for differences in behavior.

5.3 Bottleneck Transmission Infrastructure Simulation

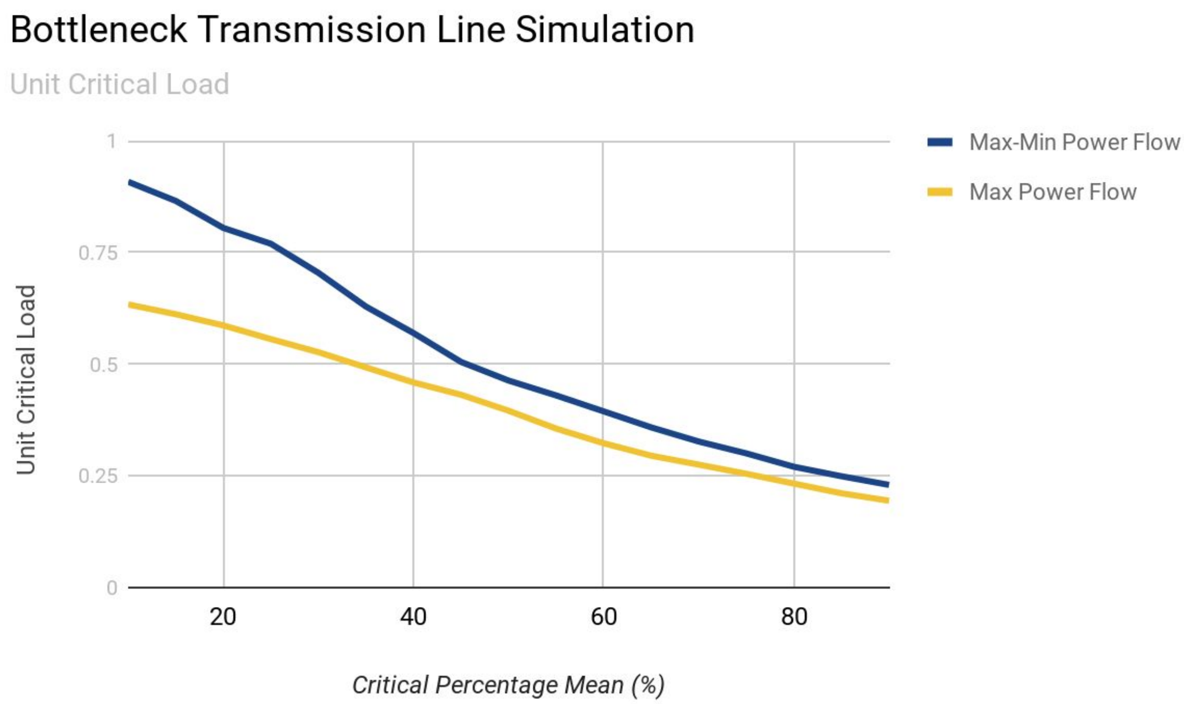

Figure 11: Unit Critical Load for an Eight Transmission Line Failure Simulation Benchmarked against Max Flow Results.

Figures 11 and 12 show the results of the Insufficient Transmission Infrastructure Simulation modified in two ways. One, the electrical limits used as flow max capacity constraints are doubled so as to not over constrain the network, and two, a max flow algorithm is run concurrently with each simulation producing its own assignment to benchmark the max-min power flow algorithm's assignment to. Only the 8 simultaneous line failures scenario is simulated.

As we can see from Figure 11, the Max-Min Power Flow, on average, widely outperforms a Max Power Flow solution in Unit Critical Load. This disparity is especially noticeable when the Critical Percentages for the load buses is under 40 percent of their total loads. This is to be expected since the Max-Min Power Flow is concerned with distributing power equitably whereas the Max Power Flow is only concerned with maximizing power usage.

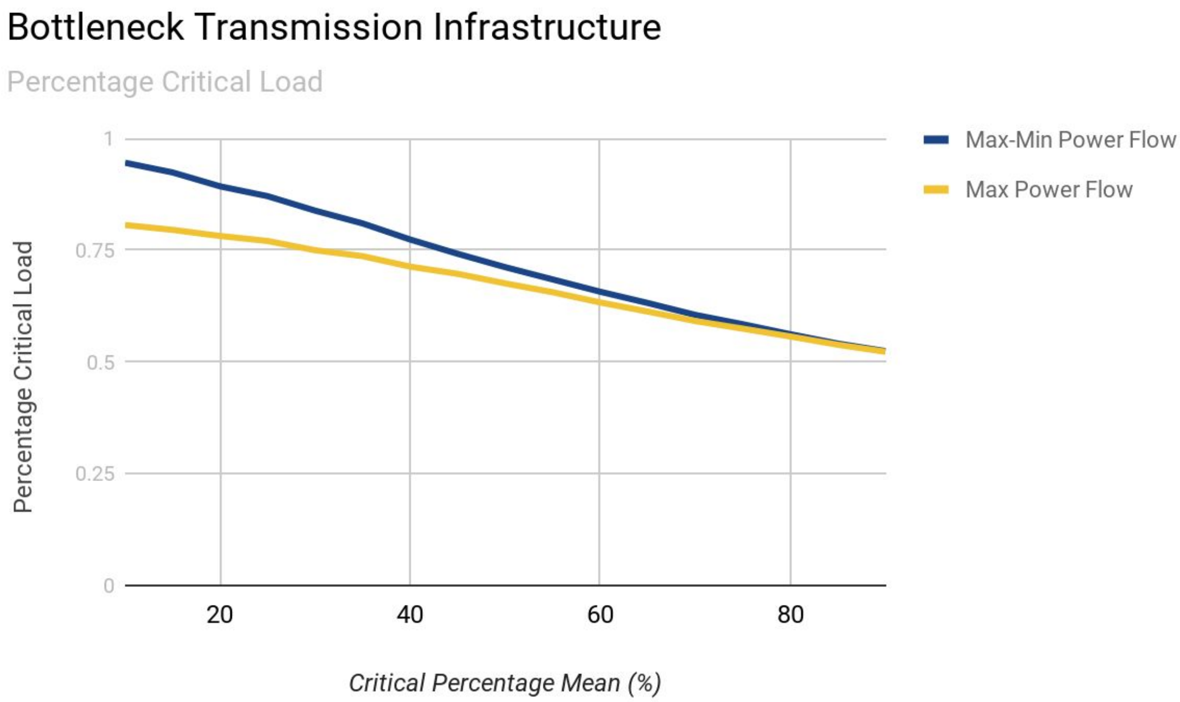

Figure 12: Percentage Critical Load for an Eight Transmission Line Failure Simulation Benchmarked against Max Flow Results.

Figure 12 presents a similar picture as Figure 11 in which the Max-Min Power Flow solution, on average, outperforms the Max Power Flow solution, but their behavior converges as the Critical Percentage Mean approaches 100 percent. In a bottleneck scenario, the Max-Min Power Flow algorithm is effective only for critical percentage means under 40%.

5.4 Runtime

Using the results of all of the simulations in this thesis (n=17000), the average runtime was .3144 seconds with a standard deviation of .057 seconds. This is well within the runtime goals for the algorithm and much faster than the distributed previous approach's 5.72 second average runtime [1]. Furthermore, all flow allocations in every simulation were feasible in the network topologies of their respective simulations meaning that a feasible solution that prevented cascading failures was found in under a second in almost all cases.

Chapter 6

Conclusion

The Max-Min Power Flow algorithm is useful in allocating power flow when large network disruptions occur during natural disasters or other emergencies. It finds a feasible solution very quickly for any network conditions to prevent cascading failures, but has mixed effectiveness depending on the type of emergency.

For attacks to generation infrastructure the algorithm is highly effective at redistributing power to satisfy critical loads in the short term. However, for damage to the transmission infrastructure enough to overload all lines simultaneously, the algorithm operates similarly to a Max-Flow power assignment in terms of satisfying critical loads. In between these two extremes, the Max-Min Power Flow assignment outperforms a Max Power Flow assignment at all critical percentage means and significantly outperforms for critical percentage means under 40%.

The speed at which the algorithm finds a feasible solution makes it useful to coordinate emergency power allocation in emergencies for power grids with buses in at least the hundreds. We conclude that the algorithm and associated protocol are useful for increasing the robustness of a power grid, but that other tools besides power flow allocation are necessary to satisfy critical loads if large parts of the transmission infrastructure are damaged.

Chapter 7

Extensions and Future Work

7.1 Adaptation to Real Conditions

This thesis uses a number of simplifications in the power grid model in order to achieve its results. Among the assumptions are the complete controllability of power injections at each bus along with its use of DC power as opposed to AC Power. In order to be adapted for real use, further work needs to be performed on the algorithm. Specifically, power flows along lines must abide by constituent relationships defined by the reactances of the transmission lines within the network. The reactances can be controlled by FACT devices [1], but the strength and location of such devices must be factored into the capacity constraints of the algorithm.

7.2 Distributed Version of the Max-Min Power Flow Algorithm

Previous research has shown that a distributed version of the Max-Flow algorithm can run on a grid with bus size 118 in on average 5.72 seconds [1]. As described inside section 3.4, the asymptotic runtime of the algorithm is $\mathbf{O(|V|^3 log(|V|))}$ where **V** is the set of buses. For small datasets with bus sizes in the hundreds this algorithm works extremely well, making it applicable to emergency power distribution to neighborhood transformers from substations and redirecting power output from a number of local power plants to distribution substations, but unlikely to be viable for the management of say a state's complete electrical power system. A distributed version of the algorithm run on a cluster consisting of a number of processors proportional to the number of buses could bring the runtime down to $\mathbf{O(|V|^2 log(|V|))}$ making it viable for a power grid with buses in the thousands.

7.3 Machine Learning approaches to Critical Load

As discussed inside the Introduction, it may not be advisable to specifically label and measure critical power loads as the information may prove a security risk. However, the use of a normal distribution to model the critical percentage may be inaccurate. Better would be able to model the critical percentage of power loads with a family of probability distribution functions such as a beta distribution that are a function of grid conditions and power consumption. Machine Learning approaches would be used to select and parameterize an appropriate probability distribution. These functions could be used to adapt the power flow allocation step of the max-min power flow algorithm to maximize the expected efficacy.

7.4 Non-Adjustable Generators

The algorithm takes into account the limit to the amount of extra power that can be generated before additional failures occur, but also assumes that generation can be adjusted downwards as much as necessary. While this may be true of conventional power plants, this is not necessarily true of all power sources. Solar panels are a common addition to homes now and many power grids allow consumers to effectively sell their surplus power generation back to the company. While technology exists to artificially reduce the amount of power generated by a solar panel, it does not exist in all homes. A next step could be to adapt the algorithm to accept set generation values.

7.5 Experimental Estimates of Critical Percentages

The concept of critical percentage is a useful tool for evaluating the efficacy of the max-min power flow algorithm. Further progress could be made with a better understanding of what the probability function for the Critical Percentage of a load looks like. This would involve pairing with a municipality, identifying critical loads and measuring those loads against the total consumption for the city to better understand what the percentages look like and how they are distributed at different times.

Chapter 8

Bibliography

[1] A. Armbruster, M. Gosnell, B. McMillin and M. L. Crow, "Power transmission control using distributed max-flow," *29th Annual International Computer Software and Applications Conference* (COMPSAC'05), Edinburgh, 2005, pp. 256-263 Vol. 2, doi: 10.1109/COMPSAC.2005.121.

[2] Chowdhury, B., & Rahman, S. (1990). A review of recent advances in economic dispatch. *IEEE Transactions on Power Systems*, *5*(4), 1248–1259. doi: 10.1109/59.99376

[3] C. S. Kong, "A general maximum power transfer theorem," in *IEEE Transactions on Education*, vol. 38, no. 3, pp. 296-298, Aug. 1995, doi: 10.1109/13.406510.

[4] Federickson, Greg N. "Fast Algorithms for Shortest Paths in Planar Graphs, with Applications." *SIAM Journal on Computing*, vol. 16, no. 6, 5 Jan. 1987, pp. 1004–1022., doi:10.1137/0216064.

[5] Fuller, T. (2019, October 10). For the Most Vulnerable, California Blackouts 'Can Be Life or Death.' *The New York Times*. https://www.nytimes.com/2019/10/10/us/california-power-outage.html

[6] G. Mazzanti, "Analysis of the Combined Effects of Load Cycling, Thermal Transients, and Electrothermal Stress on Life Expectancy of High-Voltage AC Cables," in *IEEE Transactions on Power Delivery*, vol. 22, no. 4, pp. 2000-2009, Oct. 2007, doi: 10.1109/TPWRD.2007.905547.

[7] Harks, T., Hoefer, M., Schewior, K., & Skopalik, A. (2016). Routing Games With Progressive Filling. *IEEE/ACM Transactions on Networking*, *24*(4), 2553–2562. https://doi.org/10.1109/tnet.2015.2468571

[8] Hoffman, P., & Streit, D. (2015). July 2015 United States Electricity Industry Primer. In *Department of Energy*. https://www.energy.gov/sites/prod/files/2015/12/f28/united-states-electricity-industry-primer.pdf

[9] Idaho National Laboratory. (2016). Cyber Threat and Vulnerability Analysis of the U.S. Electric Sector.

[10] Nace, D., & Pioro, M. (2008). Max-min fairness and its applications to routing and load-balancing in communication networks: a tutorial. *IEEE Communications Surveys & Tutorials*, *10*(4), 5–17. https://doi.org/10.1109/surv.2008.080403

[11] Pagani, G. A., & Aiello, M. (2013). The Power Grid as a complex network: A survey. *Physica A: Statistical Mechanics and Its Applications*, *392*(11), 2688–2700. https://doi.org/10.1016/j.physa.2013.01.023

[12] R. Christie, “Power systems test case archive: 118 bus power flow test case,” May 1993

[13] R. D. Evans and H. K. Sels, "Power limitations of transmission systems," in *Journal of the A.I.E.E.*, vol. 43, no. 1, pp. 45-51, Jan. 1924, doi: 10.1109/JAIEE.1924.6534150.

[14] S. W. Mohod and M. V. Aware, "Micro Wind Power Generator With Battery Energy Storage for Critical Load," in *IEEE Systems Journal*, vol. 6, no. 1, pp. 118-125, March 2012, doi: 10.1109/JSYST.2011.2163015.

[15] Tarjan R.E. (1986) Algorithms for maximum network flow. In: Gallo G., Sandi C. (eds) Netflow at Pisa. *Mathematical Programming Studies*, vol 26. Springer, Berlin, Heidelberg

Chapter 9

Appendix

9.1 118 Bus Generator Steady State Outputs

Bus	Power (MW)		Bus	Power (MW)		Bus	Power (MW)
1	26.48		36	10.66		74	16.93
6	0.02		40	49.32		76	22.85
10	401.87		42	40.99		80	430.84
12	85.49		46	19.04		87	3.63
15	20.88		49	193.33		89	501.84
18	13.22		54	49.54		100	231.29
19	21.58		55	32.13		103	38.25
25	193.81		56	32.56		105	5.16
26	279.76		59	149.70		107	29.03
27	9.92		61	148.41		110	7.03
31	7.25		65	352.24		111	35.24
32	14.86		66	348.86		112	36.48
34	4.88		69	453.67			

Figure 13: Table with the Steady State Generator Outputs for the 118 Bus System

9.2 Insufficient Power Generation Simulation Full Results

9.2.1 Unit Critical Loads Satisfied

%	One Fail		Four Fails		Eight Fails	
	Average	Std. Dev.	Average	Std. Dev	Average	Std. Dev.
10	1.0	0.0	1.0	0.0	.9995	.0022
15	1.0	0.0	.9997	.0017	.9978	.0053
20	1.0	0.0	.9990	.0039	.9945	.0077
25	1.0	0.0	.9979	.0046	.9903	.0084
30	1.0	0.0	.9960	.0059	.9823	.0178
35	.9995	.0022	.9903	.0113	.9793	.0155
40	.9991	.0029	.9903	.0105	.9689	.0201
45	.9980	.0040	.9860	.0138	.9621	.0312
50	.9981	.0053	.9825	.0178	.9526	.0344
55	.9970	.0063	.9765	.0247	.9503	.0364
60	.9972	.0067	.9778	.0208	.9269	.0544
65	.9943	.0106	.9706	.0247	.9187	.0649
70	.9938	.0107	.9609	.0311	.8858	.0905
75	.9936	.0118	.9600	.0411	.8750	.0948
80	.9909	.0147	.9541	.0469	.8487	.0907
85	.9906	.0149	.9511	.0544	.8437	.1110
90	.9901	.0177	.9419	.0625	.8086	.1068

Figure 14: Table of Unit Critical Loads for the Insufficient Power Generation Simulation

9.2.2 Percentage Critical Loads Satisfied

%	One Fail		Four Fails		Eight Fails	
	Average	Std. Dev.	Average	Std. Dev	Average	Std. Dev.
10	1.0	0.0	1.0	0.0	.9992	.0045
15	1.0	0.0	.9994	.0041	.9970	.0100
20	1.0	0.0	.9993	.0028	.9915	.0158
25	1.0	0.0	.9972	.0069	.9856	.0156
30	1.0	0.0	.9948	.0091	.9731	.0282
35	.9996	.0022	.9872	.0161	.9676	.0250
40	.9992	.0033	.9858	.0170	.9537	.0285
45	.9976	.0060	.9807	.0211	.9458	.0341
50	.9975	.0069	.9751	.0268	.9342	.0372
55	.9955	.0099	.9650	.0324	.9297	.0406
60	.9965	.0082	.9682	.0299	.9076	.0466
65	.9920	.0160	.9565	.0366	.9028	.0513
70	.9917	.0160	.9444	.0413	.8775	.0660
75	.9924	.0156	.9471	.0466	.8706	.0694
80	.9871	.0222	.9392	.0491	.8540	.0640
85	.9857	.0224	.9404	.0530	.8511	.0793
90	.9841	.0289	.9303	.0587	.8259	.0759

Figure 15: Table of Percentage Critical Loads for the Insufficient Power Generation Simulation

9.3 Insufficient Transmission Infrastructure Simulation Full Results

9.3.1 Unit Critical Loads Satisfied

%	One Fail		Four Fails		Eight Fails	
	Average	Std. Dev.	Average	Std. Dev	Average	Std. Dev.
10	.8408	.0272	.8383	.0385	.8253	.0385
15	.7498	.0236	.7437	.0296	.7307	.0296
20	.6617	.0208	.6560	.0214	.6409	.0214
25	.5799	.0213	.5757	.0255	.5725	.0255
30	.5146	.0185	.5061	.0222	.4972	.0222
35	.4459	.0198	.4402	.0225	.4300	.0225
40	.3870	.0195	.3778	.0216	.3691	.0216
45	.3366	.0167	.3304	.0175	.3222	.0175
50	.3006	.0149	.2971	.0176	.2896	.0176
55	.2680	.0146	.2632	.0153	.2568	.0153
60	.2330	.0132	.2260	.0161	.2218	.0161
65	.2063	.0125	.2008	.0133	.1951	.0133
70	.1892	.0074	.1856	.0118	.1802	.0118
75	.1741	.0104	.1719	.0110	.1661	.0110
80	.1551	.0130	.1487	.0162	.1468	.0162
85	.1294	.0123	.1283	.0126	.1235	.0126
90	.1037	.0116	.1031	.0106	.1052	.0106

Figure 16: Table of Unit Critical Loads for the Insufficient Transmission Infrastructure Simulation

9.3.2 Percentage Critical Loads Satisfied

%	One Fail		Four Fails		Eight Fails	
	Average	Std. Dev.	Average	Std. Dev	Average	Std. Dev.
10	.9106	.0226	.9083	.0224	.9015	.0275
15	.8733	.0175	.8693	.0231	.8619	.0227
20	.8382	.0160	.8334	.0188	.8246	.0229
25	.8027	.0119	.7994	.0154	.7894	.0181
30	.7640	.0113	.7596	.0146	.7546	.0140
35	.7228	.0127	.7207	.0133	.7170	.0152
40	.6892	.0083	.6846	.0118	.6764	.0168
45	.6546	.0087	.6484	.0134	.6427	.0142
50	.6211	.0092	.6194	.0080	.6102	.0145
55	.5928	.0058	.5865	.0097	.5826	.0113
60	.5631	.0076	.5575	.0089	.5495	.0132
65	.5347	.0065	.5305	.0093	.5213	.0134
70	.5064	.0056	.5024	.0093	.4947	.0132
75	.4816	.0049	.4770	.0073	.4723	.0098
80	.4586	.0052	.4549	.0074	.4507	.0093
85	.4370	.0059	.4339	.0070	.4292	.0093
90	.4193	.0034	.4157	.0071	.4128	.0074

Figure 17: Table of Percentage Critical Loads for the Insufficient Transmission Infrastructure Simulation

9.4 Bottleneck Transmission Infrastructure Simulation Full Results

9.4.1 Unit Critical Loads Satisfied

%	Max-Min Power		Max Power	
	Average	Std. Dev.	Average	Std. Dev
10	.9088	.0770	.6340	.0167
15	.8660	.0735	.6122	.0167
20	.8055	.0644	.5871	.0166
25	.7700	.0538	.5565	.0187
30	.7049	.0540	.5272	.0183
35	.6293	.0525	.4931	.0190
40	.5700	.0477	.4594	.0197
45	.5053	.0468	.4318	.0192
50	.4636	.0453	.3960	.0192
55	.4303	.0349	.3560	.0185
60	.3946	.0307	.3232	.0170
65	.3587	.0251	.2953	.0146
70	.3270	.0203	.2753	.0175
75	.3005	.0191	.2548	.0150
80	.2705	.0210	.2328	.0149
85	.2493	.0179	.2111	.0116
90	.2300	.0164	.1945	.0130

Figure 18: Table of Unit Critical Loads for the Bottleneck Transmission Infrastructure Simulation

9.4.2 Percentage Critical Loads Satisfied

%	Max-Min Power		Max Power	
	Average	Std. Dev.	Average	Std. Dev
10	.9461	.0352	.8067	.0252
15	.9244	.0318	.7959	.0194
20	.8933	.0253	.7822	.0188
25	.8713	.0171	.7710	.0159
30	.8389	.0205	.7503	.0183
35	.8103	.0137	.7370	.0132
40	.7743	.0169	.7136	.0173
45	.7423	.0154	.6975	.0153
50	.7124	.0152	.6761	.0143
55	.6849	.0142	.6563	.0145
60	.6574	.0142	.6334	.0131
65	.6322	.0120	.6130	.0112
70	.6056	.0159	.5918	.0142
75	.5847	.0131	.5748	.0117
80	.5624	.0112	.5573	.0112
85	.5414	.0145	.5388	.0140
90	.5246	.0125	.5231	.0126

Figure 19: Table of Percentage Critical Loads for the Bottleneck Transmission Infrastructure Simulation